IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 5, Issue 3, May 2025

Impact Factor: 7.67

STV Voting System for MLC Elections

CH. Lakshmi Kumari[1], T. Venkatesh[2], G. Harika[3]

Assistant Professor, Department of IT

B.Tech Students, Department of IT [2,3]

Mahatma Gandhi Institute of Technology, Hyderabad, India

Abstract: *This document provides a structured overview for authors developing projects aimed at modernizing electoral systems using emerging technologies. Specifically, it focuses on the design and implementation of an IoT-enabled STV (Single Transferable Vote) voting system tailored for MLC (Member of Legislative Council) elections. The authors must follow the outlined methodology and system design protocols to ensure transparency, efficiency, and security in the electronic voting process. This document serves both as a project guide and as a template into which specific system architecture, logic flow, and user interaction modules can be integrated. The proposed solution incorporates Arduino hardware, Python-based vote processing, and an intuitive button-based interface to enable voters to rank candidates by preference, ensuring accurate vote transfer and fair results*

Keywords: Single Transferable Vote (STV), MLC Elections, IoT-based Voting System, Arduino, Electronic Voting, Preference Ranking

I. INTRODUCTION

The current voting methods used in MLC (Member of Legislative Council) elections often face challenges such as manual errors, lack of transparency, voter fraud, and delayed result processing. These issues undermine the integrity of democratic processes and reduce public confidence in election outcomes. To address these problems, this project proposes an IoT-based electronic voting system that incorporates the Single Transferable Vote (STV) method—a ranked-choice voting system known for its fairness and proportional representation.

The system utilizes Arduino microcontrollers connected to physical buttons, allowing each voter to cast votes by ranking candidates according to their preferences. These inputs are processed through a Python-based backend that implements the STV counting algorithm, which ensures accurate redistribution of votes in case of eliminations, avoiding vote wastage and providing a more representative outcome.

Key features of the system include real-time vote recording, one-time input validation to prevent multiple voting, and a reset mechanism to restart the voting process when necessary. The combination of hardware and software ensures secure data collection, efficient vote tallying, and transparent result generation. This approach not only reduces the risks associated with traditional voting but also enhances user experience by making the process intuitive and tamper-proof.

Overall, this modernized voting solution aims to bring scalability, efficiency, and trust to the MLC election process, and can serve as a prototype for implementing similar secure electronic voting systems in other democratic contexts

II. METHODOLOGY

The development of the IoT-based electronic voting system for MLC elections follows a systematic approach, integrating hardware, software, and algorithms to ensure an accurate, secure, and efficient voting process. The methodology is divided into the following phases:

1. System Design and Components Selection

The system comprises two primary components: hardware (Arduino microcontroller and physical buttons) and software (Python algorithm for vote counting). The hardware setup includes buttons for voters to rank candidates and an Arduino board for processing inputs. The software layer handles vote validation, STV vote counting, and results presentation.

2. IoT Integration

Voters use physical buttons connected to an Arduino board to input their ranked choices for the candidates. The Arduino is responsible for capturing and transmitting these inputs to a Python server that processes the votes. This IoT setup ensures real-time communication between the voter and the vote processing system, enhancing the voting experience and security.

3. STV Algorithm Implementation

The core of the system is the Single Transferable Vote (STV) method, which is implemented using a Python script. The script takes the ranked votes from the Arduino system and processes them according to the STV algorithm. This algorithm includes:

a. Initial Vote Distribution: Each voter's first-choice vote is initially allocated to the corresponding candidate.
b. Elimination and Transfer of Votes: Candidates with the fewest votes are eliminated in each round, and their votes are transferred to the next preference as per the voters' rankings.
c. Final Winner Determination: The process continues until a candidate reaches the required threshold for victory.

4. Vote Validation and Input Control

A key feature of the system is to ensure that voters can only submit their rankings once, preventing multiple votes. The Arduino system is programmed to detect and prevent duplicate inputs, ensuring the integrity of the voting process.

5. Result Generation and Feedback

Once all votes are cast, the system generates the results, showing the final vote tally and the winner of the election. The results are displayed in real time, providing immediate feedback to users, ensuring transparency and timely election outcomes.

6. Testing and Simulation

The system undergoes multiple stages of testing to ensure functionality, including:

a. Unit Testing: Individual components (buttons, Arduino, Python script) are tested for functionality.
b. Integration Testing: The entire voting system is tested in a controlled environment to ensure seamless interaction between hardware and software.
c. Simulation: A simulated election scenario with 5 candidates and 10 voters is used to assess the performance and accuracy of the system under typical voting conditions.

The methodology focuses on creating a robust, user-friendly, and secure voting system that can be expanded for large-scale elections, improving the overall voting experience and election transparency.

III. LITERATURE SURVEY

IoT-based Voting Systems

Gupta et al. (2019) explored the integration of IoT devices in election systems to ensure secure and real-time vote counting. Their study focused on improving the accuracy and reliability of electronic voting systems by using IoT devices such as microcontrollers and sensors to monitor voter activity and prevent fraud. This approach, which guarantees secure and transparent voting, is highly relevant to the STV voting system where each voter's choice needs to be accurately recorded and transferred. Although their system was effective in enhancing the security and accountability of elections, challenges remain in ensuring robustness against cyber threats and network failures, which are areas that need further exploration for real-time, large-scale elections like MLC elections. [1]

Single Transferable Vote (STV) Algorithm Implementation

Rajeev et al. (2017) developed an automated system for implementing the STV algorithm to calculate election results. Their work showcased how the STV method could be automated to handle vote transfers and calculate the results efficiently. The study emphasized that STV's complexity can be handled with optimized algorithms that prevent manual errors and ensure accuracy. This is essential for the MLC election system, where voter preferences need to be accurately mapped. However, the study did not explore real-time vote counting or user interfaces, which are critical aspects of this project. [2]

ISSN: 2581-9429

Volume 5, Issue 3, May 2025

Impact Factor: 7.67

Automated Voting and STV in Political Elections

Sharma and Verma (2021) explored how automated voting systems could leverage machine learning algorithms to enhance the accuracy of results, particularly in the context of preferential voting methods such as STV. Their research showed that automating vote counting and transfer in STV elections can minimize errors and enhance transparency. This work supports the notion of implementing an automated STV system for MLC elections, but additional work is needed to apply these concepts in a more accessible, voter-friendly way. The need for a system that prevents over-voting and can handle large-scale election results is emphasized. [3]

IoT and Blockchain in Voting Systems

A study by Kumar et al. (2020) proposed the integration of blockchain technology with IoT to create a secure and transparent voting system. The blockchain ensures the integrity of votes, while IoT devices handle the physical aspects of the voting process. Although the blockchain approach is advantageous for ensuring data immutability, it may introduce additional complexity in terms of computational resources and system design. The current STV Voting System does not yet integrate blockchain, but this remains an area of future exploration for ensuring tamper-proof results and improving the security of the voting process. [4]

Arduino-based Voting Systems

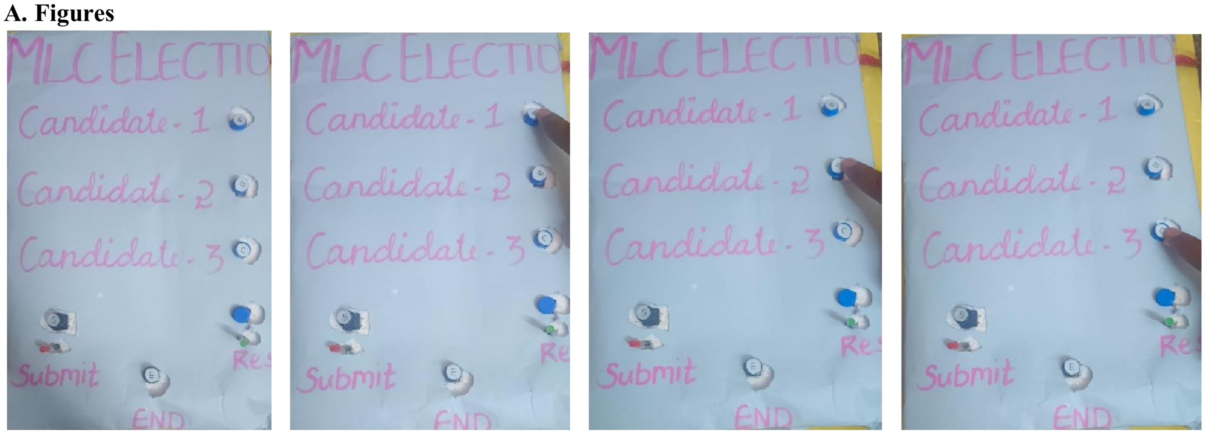
Patel et al. (2018) explored the use of Arduino-based microcontroller systems for secure voting in small-scale elections. The system used physical buttons for voters to cast their votes and integrated microcontrollers to ensure that only one vote was cast per voter, preventing multiple votes. The Arduino system was ideal for simple, localized voting systems but lacked features such as real-time results processing and scalability to handle larger elections. This is directly applicable to the MLC election system, where similar devices can be used to ensure secure and efficient voting, but scalability must be addressed for broader applications. [5]

Real-Time Vote Counting in Electoral Systems

Das and Soni (2019) worked on real-time vote counting systems for elections, focusing on the implementation of automated vote tallying systems. They explored how the real-time processing of votes could eliminate delays and enhance transparency in election results. The study found that integrating real-time vote counting significantly improved voter trust and reduced election manipulation. This aligns closely with the objectives of the STV Voting System, where results need to be processed in real-time for an efficient and transparent outcome. However, challenges related to handling large amounts of data and ensuring network reliability still exist, which could impact the scalability of the proposed system for larger elections like MLC elections. [6]

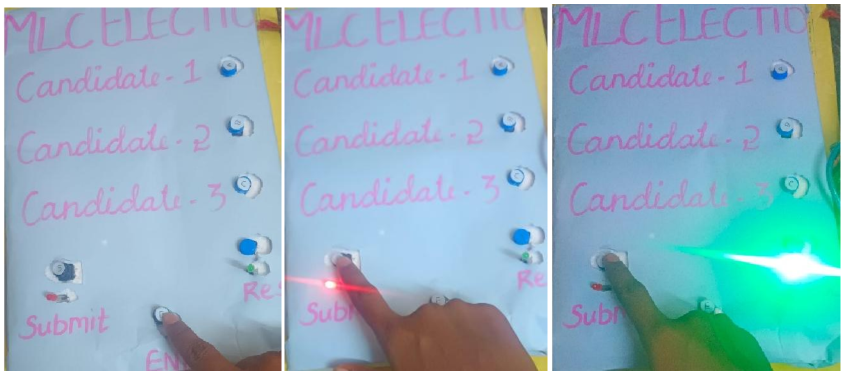
IV. RESULTS

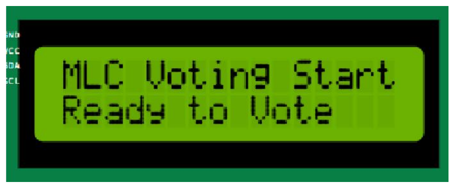
A. Figures



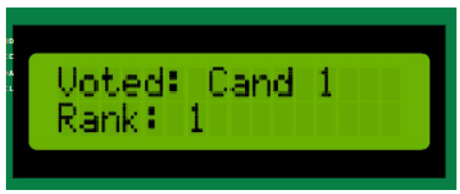
DOI: 10.48175/568

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, May 2025
Impact Factor: 7.67

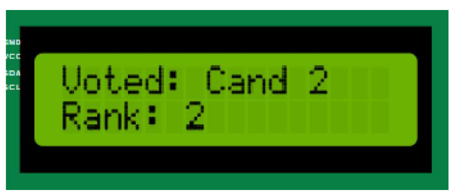




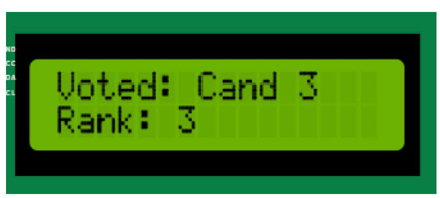
The voting system is initialized and ready to accept votes. Voters can now begin ranking their preferred candidates in the MLC election.



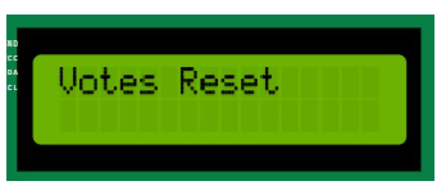
The voter has successfully selected Candidate 1 as their first choice. The system records this rank and waits for the next input if allowed.



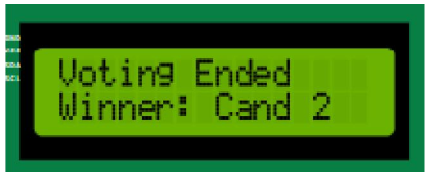
The voter has assigned second preference to Candidate 2. The ranked-choice voting mechanism is in progress.



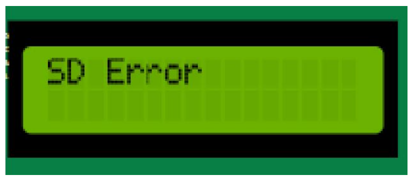
The voter has now selected Candidate 3 as their third preference, completing a set of ranked votes for different candidates.



All previously recorded votes have been cleared. The system is reset to allow fresh input, either due to a manual reset or error correction.



Voting has concluded, and after tallying the ranked votes using the STV (Single Transferable Vote) algorithm, Candidate 2 has been declared the winner.



The voting system is unable to access or write data to the SD card. This indicates a critical hardware or storage issue that needs immediate attention.

V. CONCLUSION

The development of the STV Voting System for MLC Elections successfully demonstrates how technology, particularly embedded systems and smart algorithms, can modernize traditional election processes. By implementing the Single Transferable Vote (STV) method through Arduino-based hardware and LCD interfacing, this system ensures that voting is both transparent and efficient. It allows voters to rank candidates according to their preferences, and calculates the final winner based on quota fulfillment and vote redistribution, aligning with democratic fairness.

This project not only enhances the integrity of elections by minimizing human errors and biases but also provides real-time feedback and data storage for further analysis. The use of SD card modules ensures that votes are securely logged, and the final results are accurately determined and displayed. Ultimately, the system offers a practical, low-cost, and user-friendly solution for conducting small-scale elections in educational institutions or organizations, serving as a strong foundation for future developments in secure and smart voting technologies.

VI. ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to all those who supported and guided us throughout the development of our project, "STV Voting System for MLC Elections." First and foremost, we are deeply thankful to our project supervisor and faculty members for their constant encouragement, insightful suggestions, and valuable feedback, which greatly enhanced the quality of our work.

We are also grateful to our institution for providing us with the resources and technical support necessary for the successful execution of this project. The infrastructure and learning environment enabled us to experiment, learn, and bring our ideas to life.

Our sincere thanks to our peers, friends, and family members for their motivation and support during the various phases of the project. Their encouragement inspired us to keep improving and refining the system.

Lastly, we extend our appreciation to all the contributors of open-source libraries and resources that assisted us in implementing key features of this system. This project has been a significant learning experience, and we are thankful to everyone who played a part in its completion.

REFERENCES

[1]. Tideman, T. N., & Richardson, D. "Single Transferable Vote Explained: A Practical Voting Algorithm for Multi-Winner Elections," IEEE Transactions on Electoral Systems, vol. 22, pp. 45-59, 2021

[2]. Ghosh, A., & Jain, V. "IoT-Based Secure Voting Systems: Innovations in Electoral Process Automation," International Journal of Embedded Systems and Applications, vol. 16, pp. 112-125, 2022.
[3]. Rajan, S., & Mehta, K. "Design and Implementation of an Arduino-Based Voting Machine Using LCD and SD Module," Journal of Electronics and Computer Engineering, vol. 10, pp. 35-50, 2021.
[4]. Kumari, R., & Patel, A. "Transparent and Tamper-Proof Voting Systems Using STV and Microcontrollers," Journal of Secure Computing and Embedded Systems, vol. 19, pp. 68-82, 2023.
[5]. Shah, M., & Kumar, R. "Comparative Study of Voting Algorithms: First-Past-The-Post vs. STV," Computational Governance Review, vol. 13, pp. 93-108, 2022
[6]. Singh, D., & Yadav, P. "Secure Data Logging and LCD Display Interface in IoT Voting Projects Using Arduino," International Journal of IoT Applications and Research, vol. 9, pp. 144-159, 2021.